IMPROVED CLASSIFICATION USING HIDDEN MARKOV AVERAGING FROM MULTIPLE OBSERVATION SEQUENCES

Richard I. A. Davis, Christian J. Walder and Brian C. Lovell

Intelligent Real-Time Imaging and Sensing Group,
School of Information Technology and Electrical Engineering,
The University of Queensland, Australia, 4072
{riadavis, walder, lovell}@itee.uq.edu.au

ABSTRACT

The enormous popularity of Hidden Markov models (HMMs) in spatio-temporal pattern recognition is largely due to the ability to "learn" model parameters from observation sequences through the Baum-Welch and other re-estimation procedures. In this study, HMM parameters are estimated from an ensemble of models trained on individual observation sequences. The proposed methods are shown to provide superior classification performance to competing methods.

1. INTRODUCTION

The successful application of Hidden Markov Models (HMMs) to diverse applications such as speech recognition [1, 2] and gene sequence analysis using profile HMMs [3] demonstrates the immense utility of the HMM as a workhorse for spatio-temporal pattern recognition. The usefulness of the HMM stems from the ability to learn HMM parameters through the Baum-Welch re-estimation procedure, and to provide a form of context handling in pattern recognition tasks.

In 1993, Rabiner and Juang [1] described a method where K observation sequences are used at each step of the Baum-Welch re-estimation procedure to produce a single HMM parameter estimate. This and other methods only guarantee a local maxima in model quality, leaving open the possibility of finding superior methods, especially for HMMs of specific structure.

In this paper a class of new estimation methods are proposed where the Baum-Welch re-estimation procedure is run separately to completion on the K observations and the parameters then combined to yield a single estimate. This technique expected to yield improvements because many convergence runs were used, with the results being combined using a range of methods. Such a methodology was suggested by Mackay [5] in his study of ensemble learning, but does not appear to have been investigated further. The methods used here were motivated by techniques for avoiding local minima in the context of Bayesian networks investigated by Elidan et al. [6].

Previous work by the authors [4] had investigated the performance of a range of algorithms for matching the outputs of a given HMM. Because classification is such an important application of HMMs, this paper focuses on the perfomance of the trained models on a classification task. The results are in agreement with those in [4].

2. HMM PARAMETER ESTIMATION FROM MULTIPLE OBSERVATIONS

A hidden Markov model ([1], Chapter 6) consists of a set of N nodes, each of which is associated with a set of M possible observations. The parameters of the model include an initial state

$$\pi = [p_1, p_2, p_3, ..., p_N]^T$$

with elements $p_n,\ n \in [1, N]$ which describes the distribution over the initial node set, a transition matrix

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \dots & a_{NN} \end{pmatrix}$$

with elements a_{ij} with $i, j \in [1, N]$ for the transition probability from node i to node j conditional on node i, and an observation matrix

$$\mathbf{B} = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1M} \\ b_{21} & b_{22} & \dots & b_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ b_{N1} & b_{N2} & \dots & b_{NM} \end{pmatrix}$$

with elements b_{im} for the probability of observing symbol $m \in [1, M]$ given that the system is in state $i \in [1, N]$. Rabiner and Juang denote the HMM model parameter set by $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$.

The model order pair (N, M) together with additional restrictions on allowed transitions and emissions defines the structure of the model (see figure 1 for an illustration of two different transition structures).

The Baum-Welch algorithm is an "iterative update" algorithm which re-estimates parameters of a given HMM to produce a new model which has a higher probability of generating the given observation sequence. This re-estimation procedure is continued until no more significant improvement in probability can be obtained and the local maximum is thus found. The re-estimation procedure was initiated with a random set of HMM parameters that matched the known structural constraints on the HMM (specifically Left-Right models only). This guarantees that every step of the convergence will generate a left-right model. However similar results were obtained when no transition constraints were placed upon the initial generating model.

For example, if it is known that the model is left-right and always starts in state 1, set $\pi = [1, 0, 0, \dots, 0]^T$ and A to upper-triangular. Other elements of A and B are then set to random values with the constraint that the row sums must equal unity to ensure that rows can be treated as probability mass functions. Note that elements of the matrices A and B that are set to zero will remain zero after re-estimation due to the nature of the algorithm. Hence the structure is preserved throughout the procedure. If the starting node is not known, then initially the elements of π can be set randomly with the constraint that the sum must equal unity as before. After the re-estimation runs to completion, one element, π_j say, usually becomes close to unity and all other elements usually become negligible. The index j would then represent an estimate of the starting node for this sequence.

2.1. Rabiner and Juang Method and Variants

Now consider the case where K observation sequences are known to be generated by the same HMM and the objective is to determine the HMM parameters that yield high probability of generating all K observed sequences.

Rabiner and Juang [1] proposed just such a multi-sequence training algorithm using the K observation sequences at each stage of the Baum-Welch re-estimation to iteratively update a single HMM parameter set. The re-estimation formula for this type of iterative update method is as follows (reproduced from [1]):

$$\overline{a}_{ij} = \frac{\sum_k W_k \sum_{t=1}^{T_k} \alpha_i^k a_{ij} b_j(O_{t+1}^{(k)}) \beta_{t+1}^{(k)}(j)}{\sum_k W_k \sum_{t=1}^{T_k} a_t^k(i) \beta_t^k(i)} \tag{1}$$

$$\overline{b}_{ij} = \frac{\sum_k W_k \sum_{O_t(k)=v_j} \alpha_t^k(i) \beta_t^k(i)}{\sum_k W_k \sum_{t=1}^{T_k} \alpha_t^k(i) \beta_t^k(i)} \tag{2}$$

where $W_k = 1/P_k, k \in [1 \dots k]$ is the inverse of the probability of the current model estimate generating training sequence k, evaluated using the forward algorithm [1].

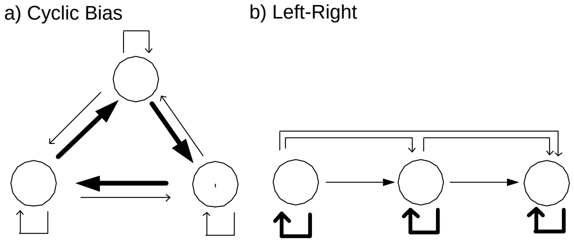

Figure 1: Cyclic and Left-Right structures showing allowed transitions. Bold arrows indicate higher probabilities.

The form of the re-estimation relation for π depends upon the model structure and is trivial in the case of left-right models (the only case covered in [1]), assuming $\pi = [1, 0, 0, \dots, 0]^T$. For other models, such as cyclic (see figure 1) one method is to run the Baum-Welch re-estimation procedure to completion on each of the K observations to estimate the starting node as above and take a simple average of the starting node distribution over all sequences. (Note that if the source model were known, it would be simple use the Viterbi algorithm [1] to estimate the starting node)

2.2. Proposed Ensemble Methods

The second approach described here is a special case of the method suggested by Mackay [5] where an ensemble of models is trained. In this paper one model is estimated for each of the K observation sequences (other approaches are possible but are not considered here). This enables the formation of K independent model estimates from the training sequences. From these the next step is to examine the efficacy of combining the independent parameter estimates using a range of simple averaging techniques of the following form:

$$\overline{a}_{ij} = \frac{\sum_k W_k a_{ij}^{(k)}}{\sum_k W_k} \tag{3}$$

$$\overline{b}_{ij} = \frac{\sum_k W_k b_{ik}^{(k)}}{\sum_k W_k} \tag{4}$$

$$\overline{\pi}_i = \frac{\sum_k W_k \pi_i^{(k)}}{\sum_k W_k} \tag{5}$$

where W_k is the weighting factor for each sequence and $\lambda^{(k)} = (A^{(k)}, B^{(k)}, \pi^{(k)})$. The quality of all model estimates is judged by the probability of that model generating an unseen set of Q test sequences from the same source as the K training sequences as described below.

An alternative involving averaging numerator and denominator before combining and normalising was also investigated, but produced estimators with performance much the same as Mackay's proposed method above.

3. METHODS INVESTIGATED AND PERFORMANCE

Simple benchmark methods:

- **Random**: Model constructed from randomly selected parameters

- **Rabiner's $W_k = 1/P_k$ Estimation** Rabiner and Juang's standard method [1] with their suggested $W_k = 1/P_k$ weighting.

- **Unit Weight**: Simple ensemble averaging over all K models,

- **Permuted Unit Weight**: Random permutations applied to the nodes of the HMM ensemble prior to paramter averaging

Method evaluation is performed using Monte Carlo simulations to produce random sources, training sequences, and test sequences. Only the performance of the models on the unseen test sequences is presented here as a measure of true learning ability.

For each model $\hat{\lambda}_k$ inferred from a sequence S_k, $k \in [1 \ldots K]$ through the Baum-Welch restimation procedure, there is an associated probability $\hat{P}_k$ of that inferred model producing the sequence S_k. Similarly, define $\hat{P}_k^{all}$ as the probability (calculated using the Forward algorithm) of that model λ_k generating all sequences S_k. Then define the following probabilities:

$$\begin{aligned}
\hat{P}_k^l &= P(l^{th}\text{training seq.} \mid \text{model for seq.}k) \\
\hat{P}_k &= P(k^{th}\text{training seq.} \mid \text{model for seq.}k) \\
\hat{P}_k^{all} &= P(\text{all } K \text{ training seq.} \mid \text{model for seq.}k) \\
&= \prod_{l=1}^{K} \hat{P}_k^l \\
\hat{P}^{all} &= P(\text{all } Q \text{ test seq.} \mid \text{estimated model}) \\
P_{true} &= P(\text{all } Q \text{ test seq.} \mid \text{true source model})
\end{aligned}$$

Note that for Rabiner and Juang's single model technique of (1) and (2), P_k is calculated during the re-estimation procedure is thus not the same as $\hat{P}_k$ above which arises after convergence of the re-estimation procedure on each of the K sequences.

It is important to distinguish between $\hat{P}_{all}$ for the test sequence set, and $\hat{P}_{all}$ for the training sequence set. This distinction is always made clear in the text of this paper.

A broader comparison of averaging methods on a sequence set approximation task may be found at [4].

4. PERFORMANCE OF NEW METHODS ON A CLASSIFICATION TASK

A study of the performance of the new HMM estimation procedure on a simple classification task was also conducted using the Bayes Net Toolbox [7]. The true and test models had 4 states, 8 observation symbols, and observations of sequence length 14 were used for both training and testing.

The trial was conducted by generating two true models and corresponding training and testing sets. Successively larger subsets of the training set were formed, and the models were trained on these subsets. Classification performance was then tested (with a testing set size of 200) on the models trained on each subset. This enabled the plotting of the number of correct classifications against training set size on our curve for each training method.

All the above steps were completed five times for five randomly chosen source models, and the mean of the curves is shown in figure 2. No structure was imposed on any of the models: all parameters were uniformly randomly chosen (an ergodic model distribution).

Unit merge training was done with random initialisation of each model before training and averaging. Another method was to randomly permute nodes relative to the nodes of other models before averaging. This experiment was motivated by the fact that we can often relabel (permute) the states of HMMs to produce equivalent models with identical probability but with quite different parameters (A, B, π). We found that if we randomly relabel in this manner before parameter averaging, we obtain HMMs that perform just as well. We believe that this is due to the fact that random initialization effectively includes a random permutation step - so a random permutation after initialization is complete will not change the average result overall. This issue will be investigated further in later work to see if small gains can be made by maximizing over the set of all relative permutations.

The random untrained models achieved about 50% correct as expected. The true generating model only achieved about 80% correct. Averaging methods approach true classification performance with about only 10 sequences; Rabiner's method required about 70 sequences to achieve the same level of performance. Hence learning for classification is more efficient for the proposed averaging methods in terms of the required number of training sequences but the asymptotic performance is similar (this was expected since it is known that the Rabiner method is quite effective at classification).

The results of this trial are consistent with the results of learning for maximum $\hat{P}_{all}$ described in the previous section.

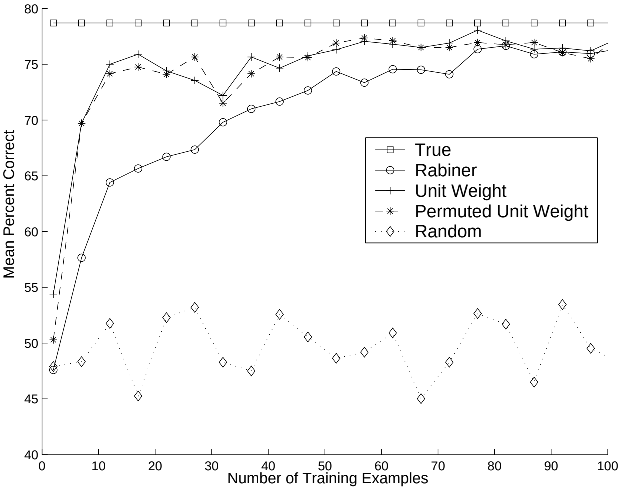

Figure 2: Classification trial ($L_{seq} = 14$).

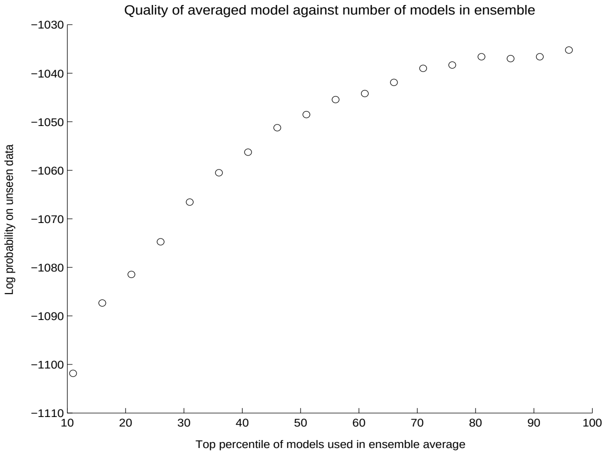

Figure 3: Mean values of $\hat{\mathbf{P}}^{\mathbf{all}}$ on test sequences when training on long training sequences ($L_{seq} = 30$).

5. EFFECT OF INCLUDING MORE MODELS

A trial was designed to study the benefits of including extra converged models in the ensemble to be averaged. Figure 3 plots the $\hat{P}_{all}$ values (defined as the log probability of generating a set of unseen test sequences against the number of models in the ensemble as additional (poorer) models are added to the ensemble. Models were combined using the unit weighting technique.

The continual improvement in performance with increasing ensemble size is a general overall trend which was consistently seen. However, in this and other cases, including a single new model or even up to 5 models in rank sequence sometimes reduces the overall model quality. It is hypothesized that this is due to the potential ambiguity for relative permutations of the HMM nodes.

6. CONCLUSIONS

The proposed Unit Weighted Ensemble averaging method for HMM parameter estimation from multiple observation sequences appears to offer significantly more probable model estimates on unseen data than the well-known method of Rabiner and Juang [1]. The classification trial demonstrated that fewer training sequences were required to achieve the same level of classification accuracy with the averaging method.

7. FUTURE WORK

Significant work remains to be done on the structure of a Windsorized algorithm and its interaction with weighting schemes and relative node permutation. It may be possible to design an algorithm which selects which method to use based upon sequence length and other parameters, thereby producing a method optimised for all sequence lengths. Finally these methods will be tested on a range of important practical problems in fields such as gene sequence analysis where precise learning (particularly from limited numbers of sample sequences) is important. Ultimately, this should lead to a comprehensive study of the parallel re-estimation problem and provide a theoretical framework to underpin future development.

8. REFERENCES

[1] L. R. Rabiner, B. H. Juang, *Fundamentals of Speech Recognition* New Jersey Prentice Hall, 1993.

[2] L. R. Rabiner, S. E. Levinson, and M. M. Sondhi, *An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition*, The Bell System Technical Journal 1035-1074, Vol. 62, No. 4, April 1983.

[3] S. R. Eddy, “Profile hidden markov models.” *Bioinformatics* Vol 14:755-763, 1998.

[4] R. I. A. Davis and B. C. Lovell, “Improved Estimation of Hidden Markov Model Parameters from Multiple Observation Sequences”, *International Congress on Pattern Recognition*, Quebec City (2002)

[5] D. J. C. Mackay, “Ensemble Learning for Hidden Markov Models”, *Technical report*, Cavendish Laboratory, University of Cambridge, 1997.

[6] G. Elidan, M. Ninio, N. Friedman, and D. Schuurmans “Data perturbation for escaping local maxima in learning.” *Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI-02)*, (2002)

[7] IRIS source for estimating Hidden Markov Models. http://www.itee.uq.edu.au/ iris/CVsource/source.html